**THEMIS – Scrum Cycle 2 Report**

Mark Acosta, Chris Beam, David Chen, Brandon Deo, Kyle Holmes, Will Paverd

Cycle Start

At the start of this cycle, we have implemented and applied much of our code, though a number of enhancements and corrections still needed to be made. A few pieces left from last cycle were still waiting to be completed, but they should take precedent and be wrapped up in short order.

a. Finalize UDSIS/Web interface implementation (#3 – 2 hours)

b. Movement clean-up and smoothing (#4 – 4 hours)

c. Re-work button implementations to use libGDX classes (#7 – 4 hours)

d. Implement objective list and progress circles (#8 – 2 hours)

e. Randomize and diversify NPC Generation, and implement character select screen (#10 – 6 hours)

f. Finalize audio implementation (#13 – 2 hours)

g. Finalize Data collection and analysis implementation (#14, 15 – 4 hours)

h. Finalize UI Implementation (2 hours)

i. Begin code review and testing (4 hours)

j. General feature implementation and enhancement (6 hours)

These tasks were assigned as follows:

Brandon: h, I, j  
 Chris: c, d  
 David: g, i  
 Kyle: e, f  
 Mark: b, e, I, j  
 Will: a, j

There should be no new resources required. Time will be freer for most team members, so implementation should be smoother and results should be forthcoming.

Testing begins this cycle, with Junit tests and code reviews beginning for approximately half of the group. Bugs will be smoothed out and eliminated as part of writing code, and since testing will begin, more bugs will be resolved as a result of code reviews and Junit testing. These fixes will be demonstrated mostly with manual testing, as well as with some automated Junit testing and code reviews.

Mid-cycle

We have made the following progress at the midpoint of the cycle:  
 a. Objectives button has been implemented, reworked and improved  
 b. Buttons have been mostly converted to use libgdx ImageButton class  
 c. Two code reviews have taken place  
 We have also hit a major obstacle, in that the game has been experience massive, crippling framerate losses. We’ve been working towards fixing the issue, but are still having trouble pinpointing what has caused the problem, which has made it quite difficult to fix said problem.